

Small Language Models for Edge and Mobile Computing: A Comparative Study of Accuracy, Latency, Energy Efficiency, and Privacy

Dr. Pankaj Kumar

Associate Professor in Computer Science, Maharani Padmavati Government College for Women, Shahzadpur
(Ambala), Haryana

Abstract

The rapid growth of generative artificial intelligence has traditionally been associated with large language models (LLMs) that require substantial computational resources, memory, and cloud infrastructure. Although these models provide strong language understanding and generation capabilities, their dependence on remote servers creates challenges related to latency, network connectivity, operational cost, energy consumption, and data privacy. Small Language Models (SLMs) provide an alternative approach by reducing model size and computational requirements while retaining useful language capabilities. This makes them particularly suitable for smartphones, tablets, Internet of Things (IoT) devices, embedded systems, and edge-computing platforms. Recent research has demonstrated that relatively compact models can provide competitive performance while being capable of local deployment. For example, Microsoft's Phi-3 Mini contains approximately 3.8 billion parameters and was specifically developed with local deployment, including smartphone execution, in mind [1]. Recent surveys also identify quantization, pruning, knowledge distillation, and architecture optimization as important techniques for enabling language models on resource-constrained devices [2,3]. This paper presents a comparative study of SLM deployment from four major perspectives: accuracy, latency, energy efficiency, and privacy. It reviews the technological foundations of edge-based language intelligence, examines model-compression techniques, and proposes an evaluation framework for comparing SLMs with larger cloud-based models. The analysis indicates that the most appropriate model for edge computing should not be selected on accuracy alone; instead, a balanced evaluation involving response quality, inference time, energy consumption, memory requirements, and privacy should be used. The paper concludes with research challenges and future opportunities for efficient, secure, and sustainable on-device intelligence.

Keywords: Small Language Models, Edge Computing, Mobile Computing, On-Device AI, Generative AI, Quantization, Energy Efficiency, Privacy, Latency, Artificial Intelligence

Date of Submission: 29-09-2026

Date of Acceptance: 09-10-2026

I. Introduction

Artificial intelligence has entered a new phase in which language models can generate text, answer questions, summarize documents, write code, and interact with users through natural language. Large Language Models have demonstrated impressive capabilities, but their deployment normally requires powerful processors, large memory resources, and high-bandwidth cloud infrastructure. These requirements limit their direct use on smartphones, IoT devices, and other resource-constrained platforms [2].

The emergence of Small Language Models changes this situation. An SLM is generally designed with a substantially smaller computational and memory footprint than a large general-purpose language model. Its objective is not simply to reduce the number of parameters, but to achieve a useful balance between model capability and deployment cost. This balance is particularly important for edge computing, where processing is performed close to the location where data is generated.

Recent research describes on-device AI as an important direction because local processing can improve responsiveness and privacy while reducing dependence on network connectivity [3]. MLCommons similarly emphasizes accuracy, latency, and energy as important dimensions for evaluating AI systems operating under resource constraints [4].

The motivation for this study is therefore based on an important question: Can small language models provide sufficient intelligence for edge and mobile applications while offering advantages in latency, energy consumption, and privacy?

This paper examines this question through a comparative framework.

Table 1. Major Differences Between Cloud-Based LLMs and Edge-Based SLMs

Parameter	Large Cloud-Based LLM	Small Edge/Mobile Language Model
Model size	Usually large	Relatively small
Primary execution	Data center/cloud	Device or edge processor
Internet dependency	Usually high	Low or optional
Response latency	Network + server dependent	Potentially low
Privacy	Data may leave device	Data can remain local
Hardware requirement	High	Lower
Energy concern	Mostly server-side	Important at device level
Customization	Cloud infrastructure required	Can be optimized for device/task
Offline operation	Limited	Strong potential
Deployment cost	Cloud/API costs	Lower recurring communication cost

The Table 1 presents the comparison because SLMs are attractive for edge environments. Cloud-based LLMs can provide broader capabilities, but communication between a user device and a remote server introduces network delay and creates additional privacy considerations. SLMs can perform inference locally, potentially allowing applications to continue working without an active Internet connection. However, smaller models generally involve a trade-off in general-purpose reasoning and knowledge coverage.

II. Small Language Models and Edge Intelligence

Small Language Models have gained importance because researchers are attempting to achieve greater computational efficiency without completely sacrificing language capability. A recent survey of SLMs identifies model architecture, training strategies, compression, pruning, and quantization as central areas of research [5].

One important example is Microsoft's Phi-3 family. The Phi-3 Mini model contains approximately 3.8 billion parameters and was trained on 3.3 trillion tokens. Microsoft reported a 69% MMLU score for Phi-3 Mini and described the model as small enough for deployment on a phone [1]. Such results demonstrate that reducing model size does not necessarily eliminate useful reasoning and language capabilities.

The objective of edge intelligence is broader than simply placing an existing cloud model on a mobile device. It requires designing an entire system around the limitations of the target hardware.

Table 2. Important Requirements for Edge-Based SLM Deployment

Requirement	Importance	Reason
Model size	Very High	Determines storage and memory requirements
RAM consumption	Very High	Mobile devices have limited available memory
Inference latency	Very High	Users expect rapid responses
Energy consumption	Very High	Directly affects battery life
Accuracy	Very High	Determines usefulness of the application
Privacy	Very High	Sensitive data can potentially remain local
Quantization support	High	Reduces memory and computational requirements
Hardware acceleration	High	Improves inference performance
Offline operation	Medium-High	Enables functionality without connectivity
Model update mechanism	High	Required for maintaining security and quality

As the table 2 shows edge deployment requires simultaneous optimization of several characteristics. A model with excellent accuracy may still be unsuitable if it consumes excessive memory or battery power. Conversely, an extremely efficient model may not be useful if its language quality is inadequate. Therefore, edge intelligence should be treated as a multi-objective optimization problem.

III. Techniques for Making Language Models Smaller

A major research challenge is reducing the computational requirements of language models while retaining acceptable accuracy. Several techniques are commonly investigated.

3.1 Quantization

Quantization represents model parameters using lower numerical precision. Instead of storing parameters using relatively high-precision formats, models may use 8-bit, 4-bit, or other reduced-precision representations. This can substantially reduce memory requirements and can improve inference efficiency on hardware that supports low-precision operations [2].

3.2 Knowledge Distillation

Knowledge distillation trains a smaller student model to reproduce useful behavior learned by a larger teacher model. The objective is to transfer important information from a computationally expensive model into a more compact model.

3.3 Pruning

Pruning removes parameters or computational components considered less important. Structured pruning can be particularly useful when the resulting architecture can be efficiently executed by the target hardware.

3.4 Architecture Optimization

Instead of shrinking a large model after training, researchers can design architectures specifically for efficiency. Efficient attention mechanisms, reduced hidden dimensions, optimized tokenization, and hardware-aware architecture design can contribute to lower inference cost.

Table 3. Model-Optimization Techniques for Edge Deployment

Technique	Main Objective	Memory Reduction	Speed Improvement	Possible Limitation
Quantization	Reduce numerical precision	High	High	Possible quality degradation
Pruning	Remove unnecessary parameters	Medium–High	Medium–High	Irregular pruning may not accelerate hardware
Knowledge Distillation	Transfer knowledge to smaller model	High	High	Requires teacher model and training
Architecture optimization	Design efficient model	High	High	Requires specialized development
Weight sharing	Reuse parameters	Medium	Medium	May reduce model flexibility
Low-rank methods	Reduce matrix complexity	Medium–High	Medium	Accuracy can vary by task

The Table 3 presents that no single compression technique is optimal for every device. Quantization is particularly attractive because it can reduce memory requirements without requiring complete model redesign. Knowledge distillation can provide a more systematic reduction in model size, while architecture-level optimization can provide larger long-term benefits when the model is designed specifically for mobile hardware. Recent research surveys emphasize quantization, pruning, and distillation as major strategies for resource-constrained deployment [2].

IV. Comparative Evaluation Framework

A meaningful comparison of SLMs should not depend on a single metric. Traditional AI evaluation often emphasizes accuracy, but edge computing introduces additional constraints.

The four primary dimensions considered in this study are:

1. **Accuracy**
2. **Latency**
3. **Energy efficiency**
4. **Privacy**

Memory footprint is also an important supporting metric.

4.1 Accuracy

Accuracy represents the ability of a model to produce useful and correct outputs. For language models, a single accuracy number is insufficient because performance can differ across question answering, summarization, reasoning, coding, and conversational tasks.

4.2 Latency

Latency measures the time between receiving an input and producing the required output. For interactive mobile applications, low latency is essential.

4.3 Energy Efficiency

Energy efficiency is especially important on battery-powered devices. MLCommons has explicitly included energy measurement in its TinyML benchmarking approach because fast inference does not automatically mean energy-efficient inference [4].

4.4 Privacy

Privacy is a major potential advantage of on-device inference. If sensitive information is processed locally, it does not necessarily need to be transmitted to a remote cloud service. However, local execution does not automatically guarantee privacy; model files, application permissions, logs, and device security must also be considered.

V. Existing Evidence and Reported Model Data

The following table presents selected data reported in published technical material rather than experimental measurements generated for this paper.

Table 4. Selected Published Evidence Relevant to Small and Edge Language Models

Model/Benchmark	Reported Information	Relevance to Edge Computing
Phi-3 Mini	3.8B parameters	Designed with local deployment in mind
Phi-3 Mini	69% MMLU reported by Microsoft	Demonstrates useful capability at smaller scale
Phi-3 Small	7B parameters	Larger than Mini but still substantially below many frontier models
Phi-3 Medium	14B parameters	Illustrates capability scaling within a compact model family
MLPerf Tiny	Measures accuracy, latency and energy	Provides a framework for resource-constrained evaluation

SLM research	Quantization, pruning and distillation widely studied	Supports practical model compression
--------------	---	--------------------------------------

Table 4 indicates that Microsoft's Phi-3 technical report provides an important example of the movement toward smaller language models. Phi-3 Mini has approximately 3.8 billion parameters and was reported to achieve 69% on MMLU [1]. MLPerf Tiny provides another important methodological reference because it treats accuracy, latency, and energy as measurable dimensions rather than focusing only on model quality [4]. These sources support the argument that edge AI should be evaluated using multiple criteria.

VI. Comparative Analysis of Accuracy, Latency, Energy and Privacy

The central argument of this paper is that SLM selection should be based on a balanced assessment rather than accuracy alone.

Table 5. Qualitative Comparative Evaluation of Edge SLM Characteristics

Model Category	Accuracy	Latency	Energy Efficiency	Privacy	Edge Suitability
Very Large Cloud LLM	Very High	Low-Medium*	Low at device level	Medium	Low
Medium Language Model	High	Medium	Medium	Medium-High	Medium
Small Language Model	Medium-High	High	High	High	High
Highly Quantized SLM	Medium-High	Very High	Very High	High	Very High
Task-Specific Tiny Model	Medium for specific task	Very High	Very High	Very High	Very High

*Latency depends on network conditions, server load, and geographic distance.

The table 5 represents a research synthesis rather than a single controlled experiment. Larger cloud models normally retain an advantage in broad knowledge and complex reasoning, but their dependence on remote infrastructure creates disadvantages for latency, connectivity, and data locality. SLMs provide a more attractive balance for applications where the task is sufficiently narrow and local processing is valuable.

VII. Proposed Experimental Methodology

A practical research study can evaluate multiple SLMs on the same mobile or edge platform. To obtain reproducible results, the hardware, software runtime, datasets, quantization level, context length, and generation settings should remain controlled.

A suitable experiment could compare three or four SLM configurations, such as a baseline model, an 8-bit version, and a 4-bit version. The models could then be evaluated using common language tasks.

Table 6. Proposed Experimental Design

Component	Proposed Setting
Models	2-4 SLM families
Hardware	Smartphone/edge development board
Precision	FP16, INT8 and INT4 where supported
Tasks	Question answering, summarization and classification
Accuracy	Task-specific evaluation score
Latency	Milliseconds per response/token
Energy	Joules per task/token
Memory	Peak RAM consumption
Privacy	Data transmission and local-processing assessment
Repetitions	Minimum 20-30 runs per configuration
Statistical analysis	Mean, median, standard deviation and confidence interval

The table 6 shows that the proposed methodology emphasizes controlled experimentation. Repeating measurements is necessary because mobile performance can fluctuate due to temperature, background applications, processor frequency scaling, and memory pressure. Both mean and median latency should be reported, while energy should preferably be measured directly using suitable hardware instrumentation.

VIII. Discussion

The comparative analysis indicates that SLMs should not be considered direct replacements for all large language models. Instead, they represent an alternative computing strategy. Large models remain advantageous for highly complex reasoning, broad knowledge retrieval, and demanding multimodal applications. SLMs are more attractive when low latency, local processing, low operating cost, offline functionality, and privacy are important.

The key finding is that model size alone is not an adequate measure of suitability. A smaller model can be preferable if it satisfies the target application's accuracy requirement while substantially reducing latency, memory consumption, and energy use.

Published work already demonstrates the feasibility of this direction. The Phi-3 Mini technical report provides evidence that a 3.8-billion-parameter language model can achieve substantial benchmark performance while being designed for local deployment [1]. At the broader edge-AI level, recent surveys emphasize the importance of compression and efficient inference techniques [2,3,5].

The major research gap is therefore not simply the creation of another small model. A more valuable contribution would be the development of standardized, reproducible multi-dimensional evaluation methods for mobile SLMs. Such evaluation should simultaneously measure accuracy, latency, energy, memory, and privacy.

IX. Conclusion

Small Language Models represent an important development in the evolution of edge and mobile artificial intelligence. Their reduced computational requirements make it possible to move a growing range of language-based AI functions from centralized cloud infrastructure toward smartphones, embedded systems, IoT devices, and edge servers.

This paper compared SLM deployment across four major dimensions: accuracy, latency, energy efficiency, and privacy. The analysis demonstrates that accuracy remains important, but it cannot be considered independently from resource consumption and deployment constraints. Low latency can improve interactive applications, energy efficiency can extend battery life, and local inference can reduce the need to transfer sensitive information to remote servers.

Current evidence from compact language models such as Phi-3 demonstrates that smaller models can provide meaningful language capabilities while targeting local execution [1]. Research on model compression further shows that quantization, pruning, knowledge distillation, and architecture optimization can support deployment in resource-constrained environments [2,5]. Benchmarking initiatives such as MLPerf Tiny reinforce the importance of evaluating accuracy, latency, and energy together [4].

The future of mobile AI is therefore likely to involve a combination of small models, hardware-aware optimization, privacy-preserving computation, adaptive inference, and selective edge-cloud collaboration. The most successful SLM systems will not necessarily be those with the smallest parameter count or the highest benchmark score. Instead, they will be systems that achieve the best balance between intelligence, responsiveness, energy consumption, security, privacy, and practical deployment requirements.

References

- [1] M. I. Abdin et al., “Phi-3 Technical Report: A Highly Capable Language Model Locally on Your Phone,” Microsoft Research, 2024. The report describes Phi-3 Mini as a 3.8-billion-parameter model and reports its performance on several academic benchmarks.
- [2] S. S. Girija, S. Kapoor, L. Arora, D. Pradhan, A. Raj, and A. Shetgaonkar, “Optimizing LLMs for Resource-Constrained Environments: A Survey of Model Compression Techniques,” IEEE COMPSAC, 2025. The work examines quantization, pruning, knowledge distillation and related optimization techniques for constrained environments.
- [3] X. Wang, Z. Tang, J. Guo, T. Meng, T. Wang, W. Jia, and others, “Empowering Edge Intelligence: A Comprehensive Survey on On-Device AI Models,” *ACM Computing Surveys*, vol. 57, no. 9, 2025. The survey discusses on-device AI with emphasis on real-time processing, resource constraints and privacy.
- [4] C. Banbury et al., “MLPerf Tiny Benchmark,” *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, 2021. The benchmark evaluates machine-learning systems using accuracy, latency and energy measurements for ultra-low-power environments.
- [5] M. Kunitovskaya and R. Mitkov, “A Survey on Small Language Models,” *Proceedings of the 15th International Conference on Recent Advances in Natural Language Processing*, 2025, pp. 807–821. The survey discusses SLM architectures, training, compression, pruning, quantization and evaluation approaches.
- [6] MLCommons, “MLPerf Tiny,” MLCommons, 2026. The benchmark initiative focuses on evaluating inference performance on extremely low-power and resource-constrained systems and emphasizes standardized measurements.
- [7] R. Wang, Z. Gao, L. Zhang, S. Yue, and Z. Gao, “Empowering large language models to edge intelligence: A survey of edge efficient LLMs and techniques,” *Computer Science Review*, vol. 57, 2025, article 100755. The study reviews techniques for enabling language-model inference on mobile, IoT and edge platforms.